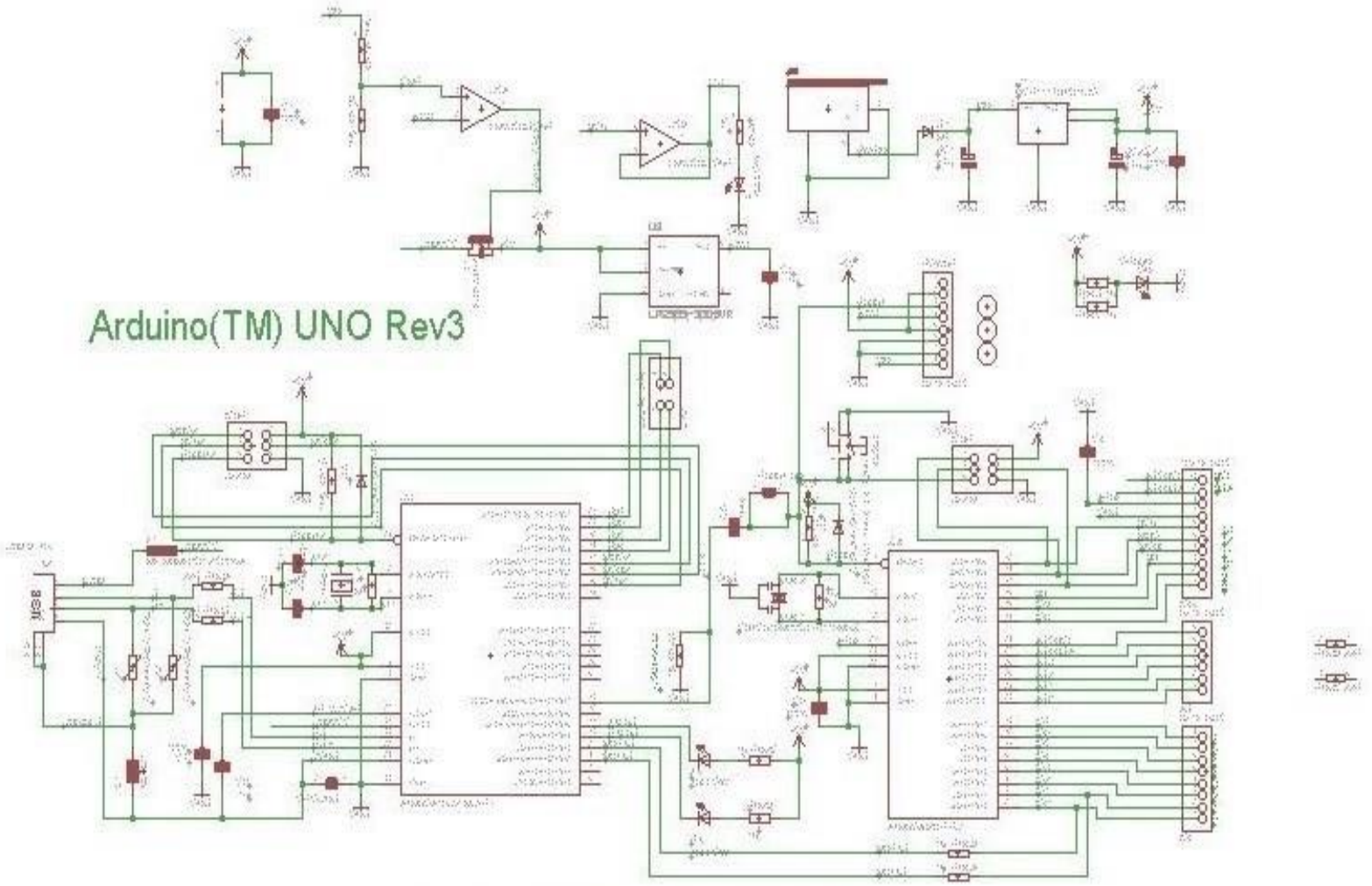
Arduino(TM) UNO Rev3

Listing Program Arduino IDE

```
#include <SoftwareSerial.h>
#include <espduino.h>
#include <rest.h>

SoftwareSerial espPort(10, 11);
ESP esp(&espPort, &Serial, 9);
REST rest(&esp);
boolean wifiConnected = false;

const int relay1 =  2;
const int relay2 =  3;
const int relay3 =  4;
const int relay4 =  5;

boolean data1 = false;
boolean data2 = false;
boolean data3 = false;
boolean data4 = false;
int loop_count = 0;

char response[266];
char buff[64];
String strId,strData,strCode;
String strData_Last1,strData_Last2,strData_Last3,strData_Last4;

void(* resetFunc) (void) = 0;
 void clearBuffer(void) {
      for (int i = 0;i<266;i++ ) {
        response[i]=0;
      }
```

```
}

void wifiCb(void* response)
{
  uint32_t status;
  RESPONSE res(response);

  if(res.getArgc() == 1) {
    res.popArgs((uint8_t*)&status, 4);
    if(status == STATION_GOT_IP) {
      Serial.println("TERHUBUNG KE WIFI");

      wifiConnected = true;
    } else {
      wifiConnected = false;
    }
  }
}

void setup() {
  pinMode(relay1, OUTPUT);
  pinMode(relay2, OUTPUT);
  pinMode(relay3, OUTPUT);
  pinMode(relay4, OUTPUT);

  digitalWrite(relay1,HIGH);
  digitalWrite(relay2,HIGH);
  digitalWrite(relay3,HIGH);
  digitalWrite(relay4,HIGH);

  Serial.begin(9600);
```

```
  espPort.begin(19200);

  esp.enable();
  delay(500);
  esp.reset();
  delay(500);
  while(!esp.ready());

  Serial.println("ARDUINO: Setup client");
  if(!rest.begin("api.thingspeak.com")) {
    Serial.println("ARDUINO: Gagal Setup client");
    while(1);
  }

  Serial.println("ARDUINO: Menghubungkan dengan Wifi");
  esp.wifiCb.attach(&wifiCb);

  esp.wifiConnect("Redmi","12345677");
  Serial.println("ARDUINO: System sudah siap!");
}

void loop() {
loop_start:

  esp.process();
  if(wifiConnected){
    char str_field1[6] , str_field2[6] , str_field3[6] , str_field4[6];

    sprintf(buff, "/channels/235564/fields/1/last");
    Serial.println(buff);
    rest.get((const char*)buff);
```

```
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
       strId = "";
       strData = "";
       strCode = "";
       getData();
       Serial.print("nilai:");
       Serial.println(strId);
       if (strId == "1" ){
         digitalWrite(relay1,LOW);
         Serial.print("high:");
         Serial.println(strId);
//       data1 = true;
       }
       else if (strId == "0"){
         digitalWrite(relay1,HIGH);
         Serial.print("LOW:");
         Serial.println(strId);
//           data1 = false;
       }
    }
     else{
//      hardReset();
//        resetFunc();
     clearBuffer();
     goto loop_start;
    }
    delay(2000);

    sprintf(buff, "/channels/235564/fields/2/last");
    Serial.println(buff);
    rest.get((const char*)buff);
```

```
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
      strId = "";
      strData = "";
      strCode = "";
      getData();
      if (strId == "1"){
        digitalWrite(relay2,LOW);
        Serial.print("high:");
        Serial.println(strId);
        data2 = true;
      }
      else if (strId == "0"){
        digitalWrite(relay2,HIGH);
        Serial.print("LOW:");
        Serial.println(strId);
        data2 = false;
      }
    }
    else{
//        hardReset();
//        resetFunc();
       clearBuffer();
       goto loop_start;
     }
    delay(2000);

    sprintf(buff, "/channels/235564/fields/3/last");
    Serial.println(buff);
    rest.get((const char*)buff);
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
      strId = "";  strData = "";  strCode = "";
```

```
      getData();                                  // GET DATA
      if (strId == "1"){
        digitalWrite(relay3,LOW);
        Serial.print("high:");
        Serial.println(strId);
        data3 = true;
      }
      else if (strId == "0"){
        digitalWrite(relay3,HIGH);
        Serial.print("LOW:");
        Serial.println(strId);
        data3 = false;
      }
    }
    else{
//        hardReset();
//        resetFunc();
      clearBuffer();
      goto loop_start;
    }
    delay(2000);

    sprintf(buff, "/channels/235564/fields/4/last");          // field x last Data
    Serial.println(buff);
    rest.get((const char*)buff);
    if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
      strId = "";  strData = "";  strCode = "";
      getData();                                  // GET DATA
      if (strId == "1"){
        digitalWrite(relay4,LOW);
        Serial.print("HIGH:");
```

```
      Serial.println(strId);
      data4 = true;
     }
     else if (strId == "0"){
      digitalWrite(relay4,HIGH);
      Serial.print("LOW:");
      Serial.println(strId);
      data4 = false;
     }
    }
    else{
//        hardReset();
//        resetFunc();
     clearBuffer();
     goto loop_start;
    }
    delay(2000);
//     loop_count++;
    Serial.println("LOOP : ");
    Serial.println(loop_count);

    if(loop_count == 6 ){
     loop_count = 0;

    if(data1) {
        dtostrf(1, 1, 1, str_field1);
    }else{
      dtostrf(0.1, 2, 1, str_field1);
    }
    if(data2) {
        dtostrf(1, 1, 1, str_field2);
```

```
    }else{
      dtostrf(0.1, 2, 1, str_field2);
    }
    if(data3) {
        dtostrf(1, 1, 1, str_field3);
    }else{
      dtostrf(0.1, 2, 1, str_field3);
    }
    if(data4) {
        dtostrf(1, 1, 1, str_field4);
    }else{
      dtostrf(0.1, 2, 1, str_field4);
    }

     sprintf(buff,
"//update?key=DL0Y8X65HX8ND1T7&field1=%s&field2=%s&field3=%s&
field4=%s",str_field1,str_field2,str_field3,str_field4);
     Serial.println(buff);
     rest.get((const char*)buff);
     Serial.println("ARDUINO: Mengirim data terbaru");
     if(rest.getResponse(response, 266) == HTTP_STATUS_OK){
       Serial.println("ARDUINO: Berhasil GET Data");
       strId = "";  strData = "";  strCode = "";
       getData();
     }
    delay(5000);
    }
  }
  else{
  }
}
```

```
void getData(){
  int i=0,j=0,k=0;
      for (i = 0; i < 10; i++){

       if((response[i] == '\r') || (response[i] == '\n')) {
       }
       else{
         strId += response[i];
       }

       if (response[i] == '\n'){
           i++;
           break;
       }
      }

      Serial.println("");
      Serial.print("ID : ");
      Serial.print(strId);

       for (j = i; j < (i+20); j++){

       if((response[j] == '\r') || (response[j] == '\n')) {
       }
       else{
         strData += response[j];
       }

       if (response[j] == '\n'){
           j++;
```

```
            break;
          }
        }

        Serial.println("");
        Serial.print("Data : ");
        Serial.print(strData);
         for (k = j; k < (j+10); k++){

          if((response[k] == '\r') || (response[k] == '\n')) {
          }
          else{
             strCode += response[k];
          }

          if (response[k] == '\n'){
               break;
          }
        }
        Serial.println("");
        Serial.print("Code : ");
        Serial.print(strCode);
        Serial.println("");
      }
  boolean hardReset() {
  String tmpData;
}
```